AN EVALUATION OF EXISTING LIGHT STEMMING ALGORITHMS FOR ARABIC KEYWORD SEARCHES

by
Brittany E. Rogerson

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

December 2008

Approved by

Advisor: Dr. Ronald E. Bergquist

Contents

Introduction to the Problem

The contemporary information retrieval environment navigated by popular search engines on the worldwide web includes not only documents on diverse subjects, but also documents in diverse languages. Many search engines even accommodate other countries and languages by forming homepages that cater to these specific populations. The popularity of Arabic language websites and documents is increasing within the realm of native speakers as the internet rises in popularity throughout the Arab world. These resources are also rising in popularity among non-native speakers as the Arabic-speaking world becomes increasingly central in world affairs. With this rise in popularity, it is necessary to reexamine the unique characteristics of the Arabic language that affect information retrieval and accommodate these characteristics with special retrieval tools such as stemming. Stemming allows a search term to focus more on the meaning of a term and closely related terms and less on specific character matches.

The Arabic language, like other Semitic languages, relies on a root system. In general, nouns and verbs are derived from a set of around 10,000 fixed roots. These roots are three, four, and sometimes five letters, and can be written in up to 62 different forms. Nouns and adjectives are derived from these verb roots using a system of rules. Additionally, Arabic uses many affixes to address grammatical points such as direct

objects, indirect objectives, and possessive pronouns.[1] For this reason, a word may include prefixes, infixes, and suffixes that contain no relevancy to the general meaning of the word itself, and only serve to elaborate on the meaning of the word in context. As a result of this root system, words on a central topic may appear in numerous forms throughout a single document. A keyword search may not return the most relevant results, and additional tools are needed to optimize a search. However, with all the possible meanings for a root, simply searching by root would force searchers to browse through a large volume of documents before finding one that meets their needs. Therefore, it is necessary to look at the research published to date and evaluate the strengths and weaknesses of each researcher's stemming and propose solutions to existing stemming algorithms to improve the Arabic language searching environment.

The Arabic Language

Before addressing the topic of stemming as a technique to improve the precision of returned documents in keyword searches, it is necessary to understand several aspects of the Arabic language that make it a particularly suitable candidate for stemming technology. The first important characteristic is shared by other Semitic languages such as Hebrew – the derivational morphology of the Arabic language. Secondly, an understanding of the use of prepositions in Arabic as juxtaposed with their use in the English language will show why simply translating English stop words, or words that are

[1] An *Affix* is a Morpheme added to a word to change its function or meaning. There are three basic ways to do this:
Prefix - by adding a morpheme to the beginning of a word.
Infix - some languages add morphemes to the middle of the word, and Arabic is one such language.
Suffix - by adding a morpheme to the beginning of a word.

omitted in keyword searches, into Arabic will not serve the same function. Lastly, the systematic way in which nouns and adjectives are derived from the consonantal verbal roots helps further display how a systematic stemming of Arabic words could improve relevancy statistics in search returns. Once the pattern of deriving nouns and verbs is clear, the motives behind stemming searching prefixes and suffixes become clear.

The Root System

Semitic languages such as Amharic, Arabic, and Hebrew use a root pattern system. The root of most words in the language is a three consonant construction with various conceptual meanings. As discussed later in this section, words including various verb forms, nouns, and adjectives are derived from these roots (Holes, 1995, p. 81). The result of the root system is that lexical sets are formed which are structurally and semantically related.

For example, the triconsonantal root ق ب ل in Arabic refers to the general idea of "meeting." In its basic form قبل the verb means he/she/it met. This verb can be manipulated into a maximum of ten standard verb forms including:

I	قبل
II	قبّل
III	قابل
IV	أقبل
V	تقبّل
VI	تقابل
VII	انقبل
VII	اقتبل
IX	قبل

X استقبل

These ten verb forms can then be manipulated in two principal ways: conjugation and pronounal (or prepositional) affixes. With conjugating verbs, it is not as necessary to try and remove the affixes because the verb is very likely to appear conjugated in the text, and most likely in a number of forms. Additional words that associate with and affix to the verbs include س or سوف used to indicate the future tense, prepositions, and enclitic pronouns or pronouns attached to the end of a verb to indicate the object of the verb.[2] Only certain prepositions join to the verb, particularly ب meaning "by" or "for" and ل also meaning "for" and sometimes meaning "in order to." Enclitic pronouns functioning grammatically as direct objects attach to the verb as suffixes as displayed in the pronoun reference chart below.

Pronouns reference chart (*Linked Pronouns,* 2006*)*

	Singular (f/m)	Plural (f/m)	Dual
1st Person	ـِي،ـنِي	ـنَا	
2nd Person	ـكِ/ـكَ /ki/ /ka/	كُنَّ-/ـكُمْ /kunna/ /kum/	ـكُمَا /kumā/
3rd Person	ـهَا/ـهُ	ـهُنَّ/ـهُم	هُـمَا

Prepositions

Having mentioned the prepositions ب and ل, it is important to note that prepositions in Arabic serve a somewhat more crucial function that those in English. A

[2] A clitic is a morpheme that shows characteristics of a word in that it has lexical meaning, but must be attached to a word. An enclitic is a clitic that is attached to the end of a word.

Deriving Verbal Nouns, Adjectives, and Place Nouns

Finally, the derivational nature of the Arabic language is essential in explaining the ability of stemming to increase the number of relevant search terms. The mostly triconsonantal roots discussed previously outline a conceptual outline for a verb in its various verb forms. From these core verbs, nouns and adjectives are derived that are semantically related to the verb. These changes are systematic and predictable because of patterns inherent in Modern Standard Arabic. Looking at the derivation of the verbal noun or gerund, adjectives, and place nouns gives a basic idea of these changes and the benefits stemming would provide for nouns and adjectives in Arabic.

For all verb forms other than form one, a rule exists for deriving the verbal noun. For example, the form two verb درّس "to teach" becomes "teaching," تدريس. In other words, a prefix ت is added along with the infix ي. The other nine forms have similar patterns involving prefixes and infixes.

Adjectives are also created using patterns as Holes describes in the fourth chapter of his work on the Arabic language (1995, p. 129). He describes the patterns using consonant and vowel structures such as CaCC, or words in which a short vowel "a" which would not be written in the Arabic script is added to the consonantal root and CaCiC in which a short vowel "a" and a long vowel symbolized by "i" but written with the Arabic letter ي are added. Examples include جميل "beautiful" from the root ج م ل and فريح "happy" from the root ف ر ح. Once again, the adjectives still clearly refer back to their roots, including only the infix ي.

A very simple derivation that occurs in the Arabic language is that of place nouns. Beginning with the base of a verb form, a noun for the place in which this action takes place is derived by adding the prefix م. For example, a مدرس is a school or a place where teaching takes place from the verb درّس to teach. A مطعم is a place where eating takes place, from the verb طعم to eat or taste. In the case of place nouns, it is clear that removing the prefix would allow semantically related words to be returned by a single keyword search.

Arabic Script

Having addressed Arabic grammar to an extent, it is important to understand the representation of the Arabic language in the written script in order to understand the components of a word that would actually be depicted in its written form. For example, as mentioned previously, short vowels are almost always excluded from Arabic texts.

Short vowels include the fatha (◌َ), Domma (◌ُ), and kasra (◌ِ). Other diacritics such as the *shadda* (◌ّ), *sikkun* (◌ْ), and tanween or double fatha, Domma, and kasra are also largely omitted. With this in mind, the keyword searches generated by users generally do not include these marks just as the texts searched by the search engines do not include them.

With this background on the Arabic language, its grammar, and its script, examining the principles of Information Retrieval will show the potential for advancements in keyword searching made possible by the root system, the semantic relevance of prepositions, and the relationship between nouns, adjectives, and the verbal roots from which they stem. Additionally, it will reveal the problematic traits of the language such as when unrelated words share a root. As described by Hayder al-Ameed and his team, "most noun, adjective, and verb stems are derived from a few thousand roots by infixing… thus, some of the most closely related forms such as singular and plural nouns are irregular and not related by simple affixation term(s)" (Al-Ameed et al., 2006, p. 944). The introductory understanding of Arabic will place the following literature in context for examining the positives and negatives of various stemming algorithm components.

Literature Review

Information Retrieval

Before broaching the topic of information retrieval for Arabic language search terms and documents, it is necessary to understand several central topics in the field of

information retrieval. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, authors of *Modern Information Retrieval* specify that prior to 1980, information retrieval almost exclusively referred to indexing texts and searching for documents (1999, p. 2). With the rise of the World Wide Web, the diversity of topics has increased as has the applicability of information retrieval to a number of fields. While the traditional topic of indexing is still applicable to the modern study of Arabic information retrieval as are several broad concepts such as recall and precision, new topics such as stemming must be addressed to provide a solid background for the introduction of contemporary issues in monolingual Arabic searching.

Indexing

Indexing is a procedure or method for accessing information that organizes the text of a document. As defined by Marie-Francine Moens in *Automatic Indexing and Abstracting of Document Texts,* "indexing commonly extracts from or assigns to the text a set of single words or phrases that function as index terms of the text" (Moens, 2000, p. 9). In other words, indexing relates groups by clustering them around words or phrases from the text. These words and phrases then become access points or identifiers of the text (Moens, 2000, p. 24). Common methods of indexing include human indexing where human beings choose the index terms based on their own knowledge and automatic or machine indexing where computer algorithms index documents.

Recall

Many studies in information retrieval choose to evaluate the performance of systems using the variable "recall." Recall is the fraction of the relevant documents existing in a corpus or set of documents where has been retrieved by a query (Baeza-Yates & Riberiro-Neto, 1999, p. 75).

Precision

Another popular performance measure and central concept to information retrieval is "precision." Baeza-Yates and Ribeiro-Neto define precision as the fraction of the retrieved documents which is relevant (1999, p.75). While recall measures the ability of a system to present all the relevant items, precision tests its ability to screen out irrelevant references (Chowdhury, 1999, p. 203). For this reason, the combination of these two variables provides a multifaceted evaluation of an information retrieval system.

Stemming

A method for improving the performance of information retrieval systems is stemming. A stem is the portion of a word which is left after the removal of its affixes. For example, the word "connect" is the stem for the words "connected," "connecting," "connections," and "connections" (Baeza-Yates & Ribeiro-Neto, 1999, p.168). Stemming takes the complex forms of a word and breaks them down to their root, under the assumption that words with the same stem are semantically related.

There are four chief methods of stemming technology: table lookup, affix removal algorithms, letter successor variety stemmers, and n-gram method (Moens, 2000, p.82).

Table lookup is the simplest method, but has the unrealistic requirement of storing stems and their related terms in a table or machine-readable dictionary. This table can quickly become very large.

Affix removal algorithms, the most common choice for Arabic language stemmers, removes affixes and employ linguistic data and knowledge about the structure of language and words to reduce terms.

Letter successor variety stemmers use data from the text to draw morphological conclusions. Looking at the sequence of letters, a stemmer is developed based on this data. Because of its instantaneous decisions, the letter successor variety stemmers are best suited for dynamic texts and collections.

Lastly, n-gram method stemming stems words based on the number of n-grams (a specified sequence of letters) they share. Terms that share letters or strings of letters are clustered into groups of related words (Moens, 2000, p.83).

Of these stemming technologies, affix removal algorithms are the most compatible with the Arabic language structure because the language's tendency to derive related terms from triconsonantal roots as previously discussed.

Arabic Information Retrieval

With the growing geo-political importance of the Arabic-speaking world and the increase in the number of Arabic websites and users, many researchers in the field of information retrieval have turned their attention to the Arabic language and its unique retrieval applications. The researchers can be divided into two principal fields: those studying monolingual Arabic searching and its implications, and those venturing into the field of cross-language searching. This paper addresses monolingual searching and

Monolingual searching studies focus on many of the same issues as English language information retrieval, including stemming and spelling standardization. Topics within the field of cross-language searching include parallel texts, automatic transliteration, Arabic synonymy sets, and localization among others. For the purpose of this study, only research in monolingual Arabic searching will aid in the evaluation of stemming algorithms. The monolingual research in the field of information retrieval will assist by providing information about the retrieval environment including users and corpora while also providing grounds for comparing the strengths of stemming vis-à-vis other technology.

Thesaurus-based Searching

One technique that provides important insight into Arabic language searching and options for broadening search terms is the use of a synonyms structure to increase recall and precision in queries. While stemming hopes to better get at the meaning of a single search term, synonym structure hopes to increase the number of search terms with a

"word sense" approach. A team of researchers at United Arab Emirates University in the city of Al-Ain have published their research concerning the application WordNet and its applications in the Arabic language (Al-Ameed et al., 2006). WordNet is a popular program in use for monolingual English language searches that uses a large lexical database of nouns, verbs, adjectives, and adverbs which are grouped into sets of cognitive synonyms each expressing a distinct concept to expand searches with a list of related search terms (*WordNet*, 2006).

In this article, the Emirati scholars advocate for the development of the program WordNet for the Arabic language. The two variables used to measure the success of such a database are precision (the percentage of retrieved documents that are relevant) and recall (the percentage of relevant documents that are retrieved). While this article does present solid evidence for the advantages of synonym digital dictionaries in information retrieval, its premise does not utilize the complexity of the Arabic language to its advantage. Words that are semantically similar in Arabic often derive from the same root. Instead of employing another interface, information retrieval scholars could instead rely on the ingenious complexity of the Arabic language itself to benefit searching. Additionally, by not discussing stemming in the lexical dictionary, the scholars would still have to treat the problem of affixations only with even more search terms were a program like WordNet employed.

Another proposal for a thesaurus-based retrieval system for Arabic is presented in the article "Empirical Studies in Strategies for Arabic Retrieval" (Xu, Fraser, & Weischedel, 2002). Here, instead of advocating a system like WordNet, the authors call

for the use of parallel corpora to construct the thesaurus by automatically generating synonyms. While this article treats cross-language searching to a large degree, it also addresses issues of spelling normalization, simple stemming, and n-gram stemming. The researchers conducted one monolingual run alongside three cross-language searches. Using the Text Retrieval Conference (TREC) Arabic corpus, the stemming techniques and spelling normalization tested improved retrieval performance by 40% and 22% respectively. In the final paragraph of their publication, the researchers suggest further research is necessary comparing their search algorithm to other Arabic stemming algorithms. While many of the stemming techniques are standard such as removing prefixes and suffixes, the algorithm of the authors tries to account for minute instances such as broken plurals, the irregular pluralizing practices which affect a small percentage of Arabic nouns. Instances like this provide more complication than benefit for monolingual Arabic searches. This paper will build on this article through comparison with other scholars' works.

Indexing for Queries

In processing text for information retrieval purposes, some scientists choose to promote the indexing of search terms instead of the stemming of search terms to make them more versatile. Like thesaurus-based searches, indexing relies on tables and databases to help provide for data for searches. Types of indexing include human and automatic indexing. Examples of automatic indexing applications for the Arabic language provide useful insight into the demands of Arabic language searchers and the performance of the technological tools that assist them.

An example of private industry applying indexing in the field of Arabic information retrieval is a program created by the company COLTEC in Egypt. The company produced the Arabic Search Plug-In (ASPI) with the purpose of channeling the complexity of the Arabic language's root system to aid in information retrieval. The company claims its personal connection to the Arabic language have allowed it to "create a unique, innovative methodology for writing Arabic language concepts and rules specifically for computer processing rather than attempting to manipulate traditional written methods that simply do not translate well to the digital world" (*ASPI*, 2008).

The tasks accomplished by the ASPI are divided into two phases: the indexing phase and the query phase. The plug-in begins by analyzing all the Arabic words and assigning specific metadata to each word during indexing. It then stores this data for future searches. When a query is run, the ASPI looks at all the previously-stored information associated with the query and provides results based on this data. While storing metadata is a good technique for giving intelligence to a system, practical issues such as where the metatags are located arise. Indexing technologies provide useful information about retrieval and different methods for increasing precision and recall of searches. In analyzing stemming technology, the goals of technologies like the ASPI provide insight into systems venturing into natural language processing.

Arabic Information Retrieval Methodology

In considering methodology practices in Arabic information retrieval, computer scientists at New Mexico State University (NMSU) produced an important work entitled

"Arabic Information Retrieval Perspectives" (Abedlali, Crowie, & Soliman, 2004). This paper discusses resources for testing Arabic information retrieval systems and methods for accelerating the development and evaluation of such systems. It shows corpora that can be used including the Linguistic Data Consortium (LDC) collection which has 869 megabytes of Arabic news articles divided into 383,872 documents from Agence France Presse (AFP). The NMSU researchers applied Zipf's Law (defined earlier as a law for the distribution of terms in a corpus that states that the most frequent term will occur twice as often as the second most frequent term, etc.) to the LDC corpus to show its completeness and representativeness. The paper goes on to discuss standard issues such as spelling normalization and transliterated proper nouns and concludes with the strengths of a newspaper corpus.

Another article of interest for Arabic Information Retrieval methodology is "Building a Modern Standard Arabic Corpus." In this article, the scholars who wrote "Arabic Information Retrieval Perspectives" discuss the various options for building a Modern Standard Arabic Corpus for testing hypotheses in information retrieval (Abdelali, Crowie, & Soliman, 2005). The authors discuss two Arabic corpora that are commonly available: the Agence France-Presse Arabic newswire from the Linguistic Data Consortium (LDC) and *Al-Hayat* newspaper collection from the European Language Resources Distribution Agency. The scholars discuss how to build a corpus using these readily available online tools. The corpora are assessed using tests such as Zipf's law and the Mandelbrot formula. While *Al-Hayat* is a pan-Arab newspaper and certainly provides a solid sampling of Modern Standard Arabic, the AFP newswire incorporates diverse

sources that will create the opportunity to test more exceptions and irregularities than *Al-Hayat*.

Stemming the Arabic Language

A lot of research has been conducted concerning stemming the Arabic language for information retrieval since stemming flows so naturally from the inherent structure of Arabic. However, a collective work evaluating these stemming methodologies against one another and incorporating the strengths of each to produce the optimal stemmer does not exist. For this reason, it is necessary to evaluate the research on Arabic stemming in order to understand the current status of stemming algorithms for the Arabic language.

Two types of stemming dominate the field: root-based stemmers and light stemmers.

Root-based stemming technology attempts to use the triconsonantal roots of the Arabic language to stem a word. Often referred to as heavy stemming, root-based stemmers remove all prefixes, infixes, and suffixes in order to deduce the semantic three-consonant root. For example, in a system applying a root-based stemmer, the term للمجاهدين or "for the freedom fighters" would remove the prefix ل meaning "for," the second ل indicating the definite article, the م indicating the "doer," the ا indicating a form three verb, and the suffix يد indicating a genitive or accusative plural human noun. These five deletions from the search term would yield a product of the root-based stemmer of ج

Light stemming, in contrast, outlines specific affixes that will be discounted in keyword searches. Light stemming can use any of the stemming methods mentioned previously (table lookup, affix removal algorithms, letter successor variety stemmers, or n-gram method) to specify certain reductions that should be made to a word.

An immediate concern for the field of stemming is overstemming which is when too much of a term is removed, causing unrelated terms to be conflated to the same stem. Light stemming hopes to compensate for this by specifying inflection morphemes, or components of words that are known to play no semantic role, only grammatical roles. However, stemmers are equally susceptible to understemming and must be thorough to ensure success.

Root-based Stemmers

One scholar currently advocating and researching the root-based stemmer is Shereen Khoja, associate professor of computer science at Pacific University. Khoja's stemmer includes lists of diacritic characters, punctuation characters, definite articles, and 168 stop words that will be used to normalize search terms. After normalization, the stemmer attempts to find the roots of the Arabic words. If no root is found, the word is left intact (Larkey & Connell, 2001, p. 565). Khoja's work was compared with a light stemmer at the 2001 session of the TREC and was shown to fall short in both precision and recall performance measures (Larkey & Connell, 2001, p. 568).

Researchers from the University of Essex built on Khoja's research (Goweder, Poesio, & De Roeck, 2004). These scholars adapted the existing root stemmer developed by Khoja and added different methods for identifying broken plurals, a weakness that Khoja had pointed out in her own system. Broken plurals are nouns that do not follow a specific pattern for pluralizing and, as a result, do not resemble their root closely in the plural form. These scholars found that practicing light stemming and adding broken plurals to a search term improved performance for information retrieval systems. In their research, light stemming with broken plural recognition outperformed standard light stemming, root stemming, and no form stemming.

Scholars Kazem Taghva, Rania Elkhoury, and Jeffrey Coombs of the University of Nevada built on Shereen Khoja's scholarship as well in their article "Arabic Stemming without a Root Dictionary" (Taghva, Elkhoury, & Coombs, 2005). Having heard of the advancements in Arabic stemming made by Larkey's team, Taghva and his team tried to improve the Khoja stemmer to make it more competitive. Mostly, the scholars eliminated the need for a dictionary to support their root stemmer, thus eliminating the intensive maintenance and system requirements associated with a dictionary of the entire Arabic language. The normalization implemented by the researchers is more conservative that that previously practiced by Khoja. For instance, the University of Nevada's Information Science Research Institute (ISRI) stemmer calls for the removal of the preliminary character و only when it precedes another و. In certain contexts, و indicated "and." However, it is also a letter of the alphabet and a viable candidate for the first letter of a word. ISRI only removes the و when it precedes another و and is

Light Stemmers

A good example of the available light stemming technology is that produced by scholars at United Arab Emirates University's software engineering department. In their article, "Arabic Light Stemmer: a New Enhanced Approach, the scholars provide an overview of five stemming algorithms that they view as an improvement from the earlier TREC-2002 algorithm (Al-Ameed et al., 2005). This article is important because it defines important terms such as "stem," "affix," and "stem-based algorithms." Additionally, the article points out degrees of stemming including light stemming and root-based approaches. The researchers present an entire list of the prefixes and suffixes that will be removed and compares this list with the affixes removed by the TREC-2002 stemming (Al-Ameed et al., 2005, p. 4).

In addition to root stemmers and affix stemmers, some scholars of information retrieval have attempted to generate rule-based Arabic stemmers. For example, scholars Ibrahim al-Kharashi and Imad al-Sughaiyer from the King Abdulaziz City for Science

and Technology in Saudi Arabia looked at techniques for analyzing Arabic morphology including the table lookup approach and a rule-based approach (2002).

Al-Kharashi and al-Sughaiyer state that the purpose of stemming algorithms and one aspect of morphological analysis techniques is to remove all possible affixes and thus reduce the word to its stem. They cite Mirko Popovic's argument published in relation to Slavic languages which shows that the effectiveness of a stemming algorithm of a given language is determined by the languages morphological complexity (Popovic & Willet, 1992). Arabic, a Semitic language derived from roots and the patterns applied to these roots, is particularly predisposed towards stemming for the reason previously stated. As the scholars' research shows, the rule-based stemmer becomes overwhelmed by the pure volume of rules involved. Therefore, the scholars also created a rule merger to cluster the rules and simplify processing. Al-Kharashi and al-Sughaiyer's research generated 1,120 rules for more than 23,000 Arabic words were investigated, producing a list of 560 merged rules. The novel research proposed in this paper would benefit from a rule-based stemmer by favoring simplicity over complex, vast stemmers that account for marginal exceptions.

Other scholarship attempts to show the benefits of light stemming algorithms over root-based algorithms with rule-based stemming algorithms. Scholars Mohammed Aljlayl and Ophir Frieder from Riyadh College of Technology and Illinois Institute of Technology respectively display their research on a light stemmer based on rules and how it out performs a root-based algorithm (Aljlayl & Frieder, 2002). Aljlayl and Frieder precisely define their definitions of normalization and stemming, listing the components

of each technique. The specifications of which elements they stem and normalize will greatly contribute to the list of items to be included in the light stemmer produced by this research. Additionally, the results put forth in their article provide statistical evidence as to why a light stemming approach was chosen over a root-based approach.

Some scholars of Information Retrieval choose a more mathematical approach that focuses on common aspects of IR such as Term Frequency (TF) and Inverse Document-Frequency (IDF). Hani Abu-Salem, Mahmoud al-Omari, and Martha W. Evans explore the idea of imposing the retrieval method over individual words of a query depending on the importance of the components of a search based on a database of words, stems, and roots that are ranked for term importance using Term Frequency and Inverse Document-Frequency (Abu-Salem, al-Omari, & Evans, 1999, p. 50). They use the variables "recall" and "precision" to evaluate the validity of their hypothesis as this study proposes to do. Most likely the material on the mathematics behind Term Frequency and Inverse Document-Frequency were not relevant to this study, but are important in understanding the precision of information retrieval. The improvement of the variables "recall" and "precision" as defined as the ratio of the number of relevant documents that are retrieved to the total number of relevant documents and the ratio of the number of relevant documents that are retrieved to the total number of retrieved documents respectively will be the goal of the proposed changes to existing stemming algorithms.

Another light stemmer was introduced by scholars Aitao Chen and Fredric Gey at the University of California at Berkeley. In their article "Building an Arabic Stemmer for

Information Retrieval," they present one monolingual Arabic run and three cross-language English-Arabic cross-language runs at the TREC of 2002 (Chen & Gey, 2002). Chen and Gey's article is of particular interest to developers of Arabic stemmers because it is extremely explicit about the stop words and affixes that it stems. Additionally, it provides the original Arabic for the stem instead of a somewhat ambiguous transliteration. In the section of their article entitled "Preprocessing," the scholars discuss the light normalization of terms that takes place. The following section covers stopwords. The elimination of stopwords will be the monumental difference between the proposals of this study and previous research by scholars such as Chen and Gey. What they call the "MT-based stemmer" discounts pronouns and prepositions, simply translating a traditional list of English stopwords into Arabic. The importance of prepositions in Arabic makes their elimination impossible; they differentiate among verb meanings as described in the Arabic language section. However, as Chen and Gey describe in section 6.2 of their article, this stemmer will discount infixes as a necessary complication and focus solely on prefixes and suffixes. For these reasons, "Building an Arabic Stemmer for Information Retrieval" will be instrumental in shaping this study.

The most paramount and influential work for Arabic light stemming is that of Leah Larkey, Lisa Ballesteros and Margaret E. Connell. In their article "Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis," Larkey et al. present a very detailed study of the Arabic morphological environment and their approaches to normalization and stemming (Larkey, Ballesteros, & Connel, 2002). The study uses recall and precision measures to evaluate their final

stemming product which they refer to as “light8-s.” The different components of Larkey’s algorithm are evaluated in the analysis section of this paper.

Objectives

This study seeks to critique existing stemming technologies in Arabic monolingual searching and propose solutions based on a combination of the findings of existing research and original contributions on the part of the researcher. The study has four steps:

1) Analyze existing normalizing and stemming technologies
2) Select positive components of existing affix removal algorithms
3) Critique negative components of existing algorithms and explain why the practice is incompatible with the Arabic language
4) Propose future research

Analysis

Selecting Text Normalization Techniques

The first step in evaluating existing stemming techniques is to create an inventory of the various components of existing normalization techniques and evaluating the choices made by previous researchers. The principal articles that outline studies including light stemming algorithms are “Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis” by Larkey’s team, “On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach” by

Mohammed Aljlayl and Ophir Frieder, and "Building an Arabic Stemmer for Information Retrieval" by Aitao Chen and Frederic Gey.

Each researcher also has a different idea of normalizing the Arabic text which must be addressed alongside the components of their affix removal algorithms. Larkey and her team choose to normalize the text by:

- Converting it to Windows Arabic encoding (CP1256)
- Removing punctuation
- Removing diacritics (short vowels)
- Removing non-letters
- Replacing all modified alefs (إ أ ٱ) with ا
- Replacing final ى with ي
- Replacing final ة (a letter usually used to indicate the feminine gender) with ه

(Larkey et al., 2002, p.278)

Aljlayl and Frieder integrate their discussion of normalization with that of stemming by listing it as steps one through four in a seven-step stemming process (Aljlayl & Frieder, 2002, p.344). Their normalization technique involves:

- Removing diacritics (short vowels)
- Replacing all modified alefs (إ أ) with ا
- Replacing final ى with ي
- Replacing the two final letters ى and ء with ئ
- Replacing the two final letters ي and ء with ئ
- Replacing final ه with ة
- Remove the prefix و if the word is three characters or longer

Lastly, Chen and Gey explain their normalization standards in the "Preprocessing" section of their study. Text normalization for the BKYMON stemmer created by Chen and Gey involves:

- Converting it to Windows Arabic encoding (CP1256)
- Considering punctuation marks to be delimiters
- Replacing all modified alefs (إ أ) with ا
- Replacing final ه with ة
- Replacing final ى with ي
- Removing the diacritic *shadda* (ّ)

None of the studies justifies its normalization techniques or makes an argument for or against certain components. Characteristics of the Arabic language itself and Arabic in word processing environments support the most basic normalization. While with stemming it can be argued that less stemming is safe and therefore better to ensure that meaning is not subtracted from the keywords, with normalization, the more normalized a text is the better. In more standardized texts, a term is more likely to find a match when unique characters, discrepancies in type, and regional preferences are eliminated. For example, in most Egyptian newspapers, the letter ي does not have the dots beneath it and appears as ى. It is possible that searchers who typically read Egyptian newspapers may enter a search term including the letter ي in its variants form.

Firstly, converting text to CP1256 is important and standard practice for Arabic language processing as it prevents font differences and other special features from interfering with one's work. It is a "code page" used to write Arabic in Microsoft Windows environments. A code page is a chart that references characters with their corresponding numerical values. These numerical values are based on American Standard Code for Information Interchange codes also known as ASCII codes. (For a chart of the Microsoft CP1256 codes, see the Appendix). This normalization practice is advisable for any Arabic language processing scholarship.

Larkey also addresses punctuation and other non-letters. She suggests removing both. The presence of punctuation and all non-letters is a logical step since the punctuation serves only a grammatical purpose rather than conveying meaning in Arabic. In English, the apostrophe can be used to convey the meaning of possession, but this rule is not present in Arabic. Possession is conveyed through a grammatical structure free of punctuation. Other non-letters such as numbers would cause too much complication as well and are best eliminated. For example, while the United States and other Western nations use Arabic numerals, most Arabic-speaking people use what they refer to as "Hindu numerals." Some newspapers and texts use the Arabic numerals out of convenience (they are always available in computer systems whereas Hindu numerals may not be present), and others remain steadfast to the Hindu numerals. Cutting out

numbers and assuming that searchers will write out important numeric values will greatly improve searching by cutting down on exceptions.

All the researchers mentioned above discuss the importance of removing diacritics from Arabic text. Diacritics in Arabic include short vowels as well as the *shadda* and *sikkun* which denote the doubling of a letter and the absence of a vowel respectively. Most texts intended for adult audiences do not include these marks and assume the read can infer them. Sometimes they are included for emphasis or for audiences of non-native or beginner speakers of Arabic. These exceptions should not be accommodated in the search environment however. The diacritics should be eliminated in favor of standardization.

The remaining normalization practices involve the standardization of Arabic letters for maximum comprehension. All Arabic letters have variant forms depending on their position in a word. For example, the letter ع may appear in three forms

1. in this form at the beginning of a word عرب,
2. this form medially العرب, and
3. finally in this form ممتع.

However, because of the ASCII coding described above and how computers process language, the variant forms do not matter. What do matter are completely different letters that are used in certain instances. Two examples specified by Larkey, Aljlayl and Frieder, and Chen and Gey are the letter ا and ي mentioned above in the

Alongside the letter ا and ي, the "*taa marbuta*" or feminine marker ة is written ه in some nations or by some Arabic literates. While Larkey's Arabic background is unknown, it is strange that she would suggest standardizing all letters written ة to ه when the latter is rarely used. Both the teams of Aljlayl and Frieder and Chen and Gey choose to normalize texts in the reverse method. With no justification supplied by Larkey, we can assume this practice is in error and suggest the widespread practice of standardizing ه to ة instead.

These normalization procedures address common practices throughout the Arabic-speaking world and in no way show preference to one region or personal preferences. By processing the text according to these normalization practices from the beginning, the resulting terms will be better candidates for searches and more likely to find matches in a variety of texts. These improvements will result in more relevant documents from keyword searches. For ready reference, the preferred normalization techniques are listed below:

- Converting it to Windows Arabic encoding (CP1256)
- Removing punctuation and other non-letters
- Removing diacritics (short vowels, *shadda*, and *sikkun*)
- Replacing all modified alefs (ٱ أ إ) with ا
- Replacing final ه with ة
- Replacing final ى with ي

Selecting Affix Removal Algorithm Components

In her study, Larkey actually discusses four separate stemmers referred to as Light1, Light2, Light3, and Light 8. The most comprehensive of these is Light8 which was also proven most effective in Larkey's research. The stems she selected for Light8 are outlined in the table below:

	Larkey et al.
Remove و	if the word is > 3 characters
Remove ال	if this leaves 2 or more characters
Remove و and ال variations	ال وال بال كال فال
Remove suffixes هـا ان ات ون ين يـه يـة ه ة ي	if this leaves 2 or more characters

(Larkey et al., 2002, p.278)

The first two rows explain the conditions under which variants of "and" and "the" will be removed. Since most Arabic words come from three letter roots, it is safe to assume that if a word is longer than three characters, an initial letter و is most likely supplemental meaning "and" and not part of the keyword itself. Similarly, with ال Larkey assumes that words will most often consists of three or more characters. There are circumstances under which only two letters may be represented, but given the statistical odds of the ا being followed by the ل at the beginning of a word, Larkey safely presumes that removing ال two or more terms appear is an acceptable practice. In the

Aljlayl and Frieder's stemming algorithm includes similar components, but could be considered to take fewer risks. Their algorithm performs the following actions:

Aljlayl & Frieder	
Remove dual and plural suffixes	ون ين ات ان
Remove masc. and fem. sing. suffixes	ة ه
Remove possessive pronouns	ي هم هن
Remove initial ل	if greater than 3 letters in length
Remove initial ب	if greater than 3 letters in length and the second character is ت
Remove initial ي	if greater than 3 letters in length and the second character is ت

(Aljlayl & Frieder, 2002, p.345)

Removing the singular and plural suffix pronouns that may be joined to keywords is necessary to increase return. Simply because a keyword does not contain the same pronoun does not mean that it is not relevant to a user's query. The same is true for possessive pronouns. Removing the initial character ل when the term is longer than three characters in length operates under the assumption as Larkey's algorithm. Because of the triconsonantal root system of Arabic, a term longer than three characters can most often be assumed to contain affixes of some sort that are not semantically relevant. Aljlayl and

The stemmer developed by Chen and Gey also identifies prefixes and suffixes to be removed. In their article, the scholars explain that they defined "one set of prefixes and one set of suffixes that should be removed based on the grammatical functions of the affixes, their occurrence frequencies among the Arabic words found in the Arabic document collection, the English translations of the affixes…" (Chen, 2002, p.635). Compiling a list of common terms and creating six lists of the one, two, and three-letter prefixes of the words and the one, two, and three-letter suffixes, they saw how many of these affixes occurred only once and how many repeated. Based on this data, the scholars compiled the following list of prefixes and suffixes to remove:

	Chen & Gey
Remove initial وال بال فال كال ولل مال ال سال لال	if longer than 5 characters
Remove initial فا كا ول وي وس سي لا وب وت وم لل با	if 4 characters or longer

Remove initial و	if 4 characters or longer
Remove initial ب ل	if 4 characters or longer
Remove final ها يـة هـم نـا مـا وا يـا نـي يـا هن كـم كـن تـم تـن يـن ان ات ون	if 4 characters or longer
Remove final ت ي ه ة	if 3 characters of longer

(Chen & Gey, 2000, p.634)

In row one, as seen with Larkey et al., Chen selects to remove variants of the definite article, those with prepositions attached. The second row of items for deletion is a mix of those with prepositions attached to part of the definite article, prepositions preceding "and," verb beginnings such as سي, and common letters to start derivatives from roots such as place nouns and alternate verb forms. Rows three and four specify "and" and two preposition prefixes that should always be removed. Rows five and six outline which of the pronoun suffixes to remove. With these specifications, Chen and Gey evaluated their light stemming algorithm.

After analyzing the scholarships of these three researchers, this study selects preferred components based on knowledge of the Arabic language, explain problems with the existing stemmers and propose solutions.

Suggestions for Optimal Stemming

For optimal stemming practices, Arabic language plug-ins and algorithmic stemmers should begin with the normalization techniques specified in the previous section. Based on the centrality of prepositions in the Arabic language for distinguishing among verb forms and verb meanings, no stop words should be considered for

monolingual Arabic searching. Other stop words in English such as “the” or “and” appear as affixes in the Arabic language instead of as separate words. Therefore, all entered search terms will be considered relevant and included in the search. Having dealt with normalization and stop words, the optimal stemming practices can be divided into five categories, those related to the articles “the” and “and,” those related to assigning gender to a word, those related to pluralizing a word, and suffixal pronoun endings.

Stemming the Definite Article

The definite article ال should be stemmed when it occurs initially and is isolated from prepositions. The word resulting from the stemming process must be at least two characters in length to be acceptable. The nature of the Arabic language necessitates that prepositions be joined to the definite as in بالسيارة where the preposition ب joins to the definite article. Since these prepositions could be instrumental in deciphering the meaning of the key term in its context, only isolated incidents of ال will be stemmed. The one exception to this stemming rule will be the conjoined characters وال meaning “and the.” Neither of these phonemic morphemes is seen as relevant to the meaning of the key term. The stemming rule mandates that the sequence of characters be greater than three in length to ensure that the stemmer is not eliminating portions of a phoneme. Given that over ninety percent of Arabic words are derived from triconsonantal roots, the words with ال that are greater than three characters in length are highly unlikely to lost phonemically relevant units. For example, the words

الكتاب "the book,"

الحب "the love," and

الحرية "the freedom"

all lose the definite article, but retain all relevant characters with these stemming rules. These rules are conservative and based on the removal of grammatical units only instead of units that add meaning to keywords. When compared with existing algorithms, this proposed practice most closely aligns with Larkey's practice in its specification of word length, but differs from those of both Larkey and Chen and Gey in excluding most variants of the definite article that appear when a preposition or conjunction is attached. These units could be lexically relevant and should not be excluded.

Stemming the Feminine Marker

The feminine marker ة is most often used to assign gender to a noun or adjective rather than to differentiate the meaning. Examples include the transformation of مدرس to مدرسة or جميل to جميلة. Certain words simply have an innate gender and therefore take the feminine marker such as كرة or كلمة as in languages such as French or Spanish. For this component of the algorithm, Larkey's specifications in her Light8 stemmer successfully stem the feminine marker without subtracting the meaning of the keywords. Larkey suggests that ة and its variant ية be removed if the removal leaves two or more characters after processing. This conservative rule for stemming the suffixal feminine marker will contribute to the improvement of recall and precision by discounting the

Stemming Plural Suffixes

The Arabic language consists of a variety of suffixes that serve to make a singular noun, verb, or adjective into the dual or plural. These dual and plural endings can change with case and position in the sentence. If the dual or plural ending is added to a word that is in the nominative case, it will be ان for the masculine dual and تان for the feminine dual and ون for the human masculine plural and ات for the non-human feminine plural. In the genitive and accusative cases, these endings change to تين ين and ين while ات remains the same. An exception to these suffixes exists. When the grammatical structure الإضافة is used, two nouns are placed consecutively forming the meaning "of." For example, when the word كتاب "book" precedes المدرس "the teacher," the resulting meaning is "the book of the teacher" or "the teacher's book." If a dual or plural noun represents the first noun

For the dual endings and feminine plural ending, Larkey is too conservative and does not take into account the genitive and accusative endings or the feminine dual ending. She only lists ان and ignores تان and تين as do Aljlayl and Frieder and Chen and Gey. These endings should be removed by the same logic as ان since the dual endings in no way contribute to the lexical value of the search terms.

Stemming Suffixal Pronoun Endings

The Arabic language attaches pronouns indicating possession or acting as the direct object directly to the nouns or verbs with which they are associated. For example, "your (singular) book" would be كتابك or "I love you" would be أحبك. For all persons except first person singular, the suffix is the same whether it attaches to a verb or a noun. All of these suffixes should be stemmed in instances where the sequence is four characters or longer. The suffixes are: هن هم نا كما كم ها ه ك ي ني. Aljlayl and Frieder only remove هم and هن and ignore the other options. Chen and Gey go further with the pronouns, but include questionable entries such as تم and كن which are more often parts of triconsonantal roots and less often pronouns. Larkey is too conservative and ignores all of the suffixal pronouns except for ه and ها. The best choice is to take the moderate path of stemming items which are almost always grammatical units.

The following chart summarizes the components of the stemming process:

Remove initial وال ال	if this leaves 2 or more characters
Remove final ني ي ك ه ها نا كما كم هم هن	if 4 characters or longer
Remove final ون تـــان تيـــن ان يـــن و ي ات يـــن	if this leaves 2 or more characters
Remove final يـــة ة	if this leaves 2 or more characters

Items Excluded from the Stemming

Several of the items in the stemming algorithms attempt to stem too much and may subtract lexically relevant units from keywords. As previously mentioned, both Larkey and Chen and Gey attempt to stem variants of ال which subtract crucial

Aljlayl et al. and Chen et al. take several steps in an attempt to stem verbs. While certainly having a stemmed verb would increase the potential for matches in a query, verbs vary so extremely over their ten forms and 3 tenses that stemming is almost impossible without removing lexically relevant portions of other words. Aljlayl stems ي when it occurs as an initial letter since present tense verbs begin with ي when not preceded by a preposition. However, many words in the Arabic language also begin with ي as do proper names. This effort, while well intentioned, is most likely impossible given the lexicon of the Arabic language. Chen and Gey attempt to stem verbs and verb particles as well with their algorithm. They implement an algorithm that removes the initial letter س and سي, the beginning of future tense verbs. Once again, many words in the language begin with these letters. Chen and Gey's constraint that the word must be longer than four characters does nothing to curb the elimination of the preliminary letters of numerous, numerous Arabic words.

With these suggestions and the subtractions, a superior, conservative yet thorough, stemmer could be constructed to enhance the performance of Arabic language keyword search engines and improve recall and precision. With these steps, the Arabic-speaking world could have better access to Arabic language materials and thereby elevate the status of the Arab world in the fields of technology and research.

Future Research

Several constraints exist for the field of Arabic information retrieval at present including few options for search engines, computers without Arabic language capabilities, and a lack of interest in elevating the search capabilities of non-Western languages given the dominance of English on the World Wide Web. The most disruptive restraint to stemming research, however, is a lack of wildcard functionality on almost all Arabic language search engines. Some provide wildcard searching at a word level. For example in English, searching "the cat ** hat" would return "the cat in the hat," "the cat wore the hat" or other results where two words separated "cat" and "hat." However, they do not allow for letter-level wildcards such as "c*t" returning "cat," "cut," and "cot." Without this technology, the popular use of stemming algorithms in search engines as either plug-ins or interfaces cannot achieve popularity or benefit large populations. The Arabic language information retrieval world could head down one of two paths. Either search engines progress as English language engines have and increase their functionality to satisfy users, or users will choose transliterated Arabic using Latin characters or English language searching, for bilingual users. In order to preserve the lingual diversity of the physical and electronic environments and promote the technological and academic prosperity of other parts of the world, research must continue to level the information retrieval field so that monolingual searches in all languages return relevant results efficiently. With stemming and wildcard functionalities, this equality is possible.

References

Abdelali, Ahmed, Cowie, Jim, & Soliman, Hamdy S. (2004). Arabic Information Retrieval Perspectives. *Proceedings of JEP-TALN 2004 Arabic Language Processing*.

Abdelali, Ahmed, Cowie, James and Soliman, Hamdy S. (2005). Building a Modern Standard Arabic Corpus. *Workship on Computational Modeling of Lexical Acquisition.*

Abu-Salem, Hani, al-Omari, Mahmoud , & Evens, Martha W. (1999). Stemming Methodologies over Individual Query Words for an Arabic Information Retrieval System. *Journal of the American Society for Information Science, 50.*

Al-Ameed, Hayder K., al-Ketbi, Shaikha O., al-Kaabi, Amna A., al-Shebli, Khadija S., al-Shamsi, Nalia F., al-Nuaimi, Noura H., & al-Muhairi, Shaikha S. (2006). Arabic Search Engines Improvement: a New Approach using Search Key Expansion Derived from Arabic Synonyms Structure. *AICCSA '06: Proceedings of the IEEE International Conference on Computer Systems and Applications*, 944-951.

Al-Ameed, Hayder K., al-Ketbi, Shaikha O., Al-Kaabi, Amna A., al-Shebli, Khadija S. , al-Shamsi, Nalia F., al-Nuaimi, Noura H., & al-Muhairi, Shaikha S. (2005). Arabic Light Stemmer: a New Enhanced Approach. *Proceedings of the Second International Conference on Innovations in Information Technology*.

Aljlayl, Mohammed & Frieder, Ophir. (2002). On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach. *Proceedings of the eleventh international conference on Information and knowledge management,* 340-347.

Al-Kharashi, Ibrahim A. & al-Sughaiyer, Imad A. (2002). Rule Merging in a Rule-Based Arabic Stemmer. *Proceedings of the 19th International Conference on Computational Linguistics- Volume 1,* 1-7.

ASPI Arabic Search Plug-In. (2008). COLTEC. Retrieved 29 Jul 2008 from http://www.coltec.net/Portals/0/COLTEC_PDFs /COLTEC_PlugIn_NEW.pdf.

Baeza-Yates, Ricardo and Riberiro-Neto, Berthier. (1999). *Modern Information Retrieval*. New York: ACM Press.

Chen, Aitao & Gey, Fredric. (2002). Building an Arabic Stemmer for Information Retrieval. *TREC 2002,* 631-639.

Chowdhury, G. G. (1999). *Introduction to Modern Information Retrieval.* London: Library Association Publishing.

Goweder, Abduelbaset, Poesio, Massimo & De Roeck, Anne. (2004). Broken Plural Detection for Arabic Information Retrieval. *Annual ACM Conference on Research and Development in Information Retrieval: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval,* 566-567.

Holes, Clive. (1995). *Modern Arabic: Structures, Functions, and Varieties,* New York: Longman Publishing.

Larkey, Leah S. & Connell, Margaret E. (2001). Arabic Information Retrieval at UMass in TREC-10. *TREC 2001,* 562-570.

Larkey, Leah, Ballesteros, Lisa, & Connel, Margaret E. (2002). Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis. *SIGIR 2002,* 275-282.

Linked Pronouns. (2006). Retrieved September 8, 2008, from # Arabic Wiki Area Web site: http://www.mesiti.it/arabic/wiki/wiki.asp?db=wikiasp&o=LinkedPronouns#1

Moens, Marie-Francine. (2000). *Automatic Indexing and Abstracting of Document Texts.* Boston: Kluwer Academic Publishers.

Popovic, M. & Willet, P. The effectiveness of stemming for natural-language access to Slovene textual data. .(1992). *Journal of the American Society for Information Sciences*, 384-390.

Taghva, Kazem, Rania Elkhoury, & Coombs, Jeffrey. (2005). Arabic Stemming without a Root Dictionary. *Proceedings of the International Conference on Information Technology: Coding and Computing.*

Wehr, Hans. (1980). *A Dictionary of Modern Written Arabic.* London: MacDonald and Evans Ltd., 798.

WordNet: a lexical database for the English language . (2006). Retrieved September 21, 2008, from Princeton University Web site: http://wordnet.princeton.edu/

Xu, Jinxi, Fraser, Alexander , & Weischedel, Ralph. (2002). Empirical Studies in Strategies for Arabic Retrieval. *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval,* 269-274.

Appendix

Windows 1256

(2005). *Windows 1256*. Retrieved October 29, 2008, from Global Development and Computing Portal website: http://www.microsoft.com/globaldev/reference/sbcs/1256.mspx.

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	NUL 0000	STX 0001	SOT 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
10	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
20	SP 0020	! 0021	" 0022	# 0023	$ 0024	% 0025	& 0026	' 0027	(0028	) 0029	* 002A	+ 002B	, 002C	- 002D	. 002E	/ 002F
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	: 003A	; 003B	< 003C	= 003D	> 003E	? 003F
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C	] 005D	^ 005E	_ 005F
60	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	| 007C	} 007D	~ 007E	DEL 007F
80	€ 20AC	پ 067E	‚ 201A	ƒ 0192	„ 201E	… 2026	† 2020	‡ 2021	ˆ 02C6	‰ 2030	ٹ 0679	‹ 2039	Œ 0152	چ 0686	ژ 0698	ڈ 0688
90	گ 06AF	‘ 2018	’ 2019	“ 201C	” 201D	• 2022	– 2013	— 2014	ک 06A9	™ 2122	ڑ 0691	› 203A	œ 0153	ZWNJ 200C	ZWJ 200D	ں 06BA
A0	NBSP 00A0	، 060C	¢ 00A2	£ 00A3	¤ 00A4	¥ 00A5	¦ 00A6	§ 00A7	¨ 00A8	© 00A9	ھ 06BE	« 00AB	¬ 00AC	- 00AD	® 00AE	¯ 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	؛ 061B	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	؟ 061F
C0	ہ 06C1	ء 0621	آ 0622	أ 0623	ؤ 0624	إ 0625	ئ 0626	ا 0627	ب 0628	ة 0629	ت 062A	ث 062B	ج 062C	ح 062D	خ 062E	د 062F
D0	ذ 0630	ر 0631	ز 0632	س 0633	ش 0634	ص 0635	ض 0636	× 00D7	ط 0637	ظ 0638	ع 0639	غ 063A	ـ 0640	ف 0641	ق 0642	ك 0643
E0	à 00E0	ل 0644	â 00E2	م 0645	ن 0646	ه 0647	و 0648	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ى 0649	ي 064A	î 00EE	ï 00EF
F0	ً 064B	ٌ 064C	ٍ 064D	َ 064E	ô 00F4	ُ 064F	ِ 0650	÷ 00F7	ّ 0651	ù 00F9	ْ 0652	û 00FB	ü 00FC	LTR 200E	RTL 200F	ے 06D2

```
00 = U+0000 : NULL
01 = U+0001 : START OF HEADING
02 = U+0002 : START OF TEXT
03 = U+0003 : END OF TEXT
04 = U+0004 : END OF TRANSMISSION
05 = U+0005 : ENQUIRY
06 = U+0006 : ACKNOWLEDGE
07 = U+0007 : BELL
08 = U+0008 : BACKSPACE
09 = U+0009 : HORIZONTAL TABULATION
0A = U+000A : LINE FEED
0B = U+000B : VERTICAL TABULATION
0C = U+000C : FORM FEED
0D = U+000D : CARRIAGE RETURN
0E = U+000E : SHIFT OUT
0F = U+000F : SHIFT IN
10 = U+0010 : DATA LINK ESCAPE
11 = U+0011 : DEVICE CONTROL ONE
12 = U+0012 : DEVICE CONTROL TWO
13 = U+0013 : DEVICE CONTROL THREE
14 = U+0014 : DEVICE CONTROL FOUR
15 = U+0015 : NEGATIVE ACKNOWLEDGE
16 = U+0016 : SYNCHRONOUS IDLE
17 = U+0017 : END OF TRANSMISSION BLOCK
18 = U+0018 : CANCEL
19 = U+0019 : END OF MEDIUM
1A = U+001A : SUBSTITUTE
1B = U+001B : ESCAPE
1C = U+001C : FILE SEPARATOR
1D = U+001D : GROUP SEPARATOR
1E = U+001E : RECORD SEPARATOR
1F = U+001F : UNIT SEPARATOR
20 = U+0020 : SPACE
21 = U+0021 : EXCLAMATION MARK
22 = U+0022 : QUOTATION MARK
23 = U+0023 : NUMBER SIGN
24 = U+0024 : DOLLAR SIGN
25 = U+0025 : PERCENT SIGN
26 = U+0026 : AMPERSAND
27 = U+0027 : APOSTROPHE
28 = U+0028 : LEFT PARENTHESIS
29 = U+0029 : RIGHT PARENTHESIS
2A = U+002A : ASTERISK
2B = U+002B : PLUS SIGN
2C = U+002C : COMMA
2D = U+002D : HYPHEN-MINUS
2E = U+002E : FULL STOP
2F = U+002F : SOLIDUS
30 = U+0030 : DIGIT ZERO
31 = U+0031 : DIGIT ONE
32 = U+0032 : DIGIT TWO
33 = U+0033 : DIGIT THREE
34 = U+0034 : DIGIT FOUR
35 = U+0035 : DIGIT FIVE
36 = U+0036 : DIGIT SIX
37 = U+0037 : DIGIT SEVEN
38 = U+0038 : DIGIT EIGHT
39 = U+0039 : DIGIT NINE
3A = U+003A : COLON
3B = U+003B : SEMICOLON
3C = U+003C : LESS-THAN SIGN
3D = U+003D : EQUALS SIGN
3E = U+003E : GREATER-THAN SIGN
3F = U+003F : QUESTION MARK
```

```
40 = U+0040 : COMMERCIAL AT
41 = U+0041 : LATIN CAPITAL LETTER A
42 = U+0042 : LATIN CAPITAL LETTER B
43 = U+0043 : LATIN CAPITAL LETTER C
44 = U+0044 : LATIN CAPITAL LETTER D
45 = U+0045 : LATIN CAPITAL LETTER E
46 = U+0046 : LATIN CAPITAL LETTER F
47 = U+0047 : LATIN CAPITAL LETTER G
48 = U+0048 : LATIN CAPITAL LETTER H
49 = U+0049 : LATIN CAPITAL LETTER I
4A = U+004A : LATIN CAPITAL LETTER J
4B = U+004B : LATIN CAPITAL LETTER K
4C = U+004C : LATIN CAPITAL LETTER L
4D = U+004D : LATIN CAPITAL LETTER M
4E = U+004E : LATIN CAPITAL LETTER N
4F = U+004F : LATIN CAPITAL LETTER O
50 = U+0050 : LATIN CAPITAL LETTER P
51 = U+0051 : LATIN CAPITAL LETTER Q
52 = U+0052 : LATIN CAPITAL LETTER R
53 = U+0053 : LATIN CAPITAL LETTER S
54 = U+0054 : LATIN CAPITAL LETTER T
55 = U+0055 : LATIN CAPITAL LETTER U
56 = U+0056 : LATIN CAPITAL LETTER V
57 = U+0057 : LATIN CAPITAL LETTER W
58 = U+0058 : LATIN CAPITAL LETTER X
59 = U+0059 : LATIN CAPITAL LETTER Y
5A = U+005A : LATIN CAPITAL LETTER Z
5B = U+005B : LEFT SQUARE BRACKET
5C = U+005C : REVERSE SOLIDUS
5D = U+005D : RIGHT SQUARE BRACKET
5E = U+005E : CIRCUMFLEX ACCENT
5F = U+005F : LOW LINE
60 = U+0060 : GRAVE ACCENT
61 = U+0061 : LATIN SMALL LETTER A
62 = U+0062 : LATIN SMALL LETTER B
63 = U+0063 : LATIN SMALL LETTER C
64 = U+0064 : LATIN SMALL LETTER D
65 = U+0065 : LATIN SMALL LETTER E
66 = U+0066 : LATIN SMALL LETTER F
67 = U+0067 : LATIN SMALL LETTER G
68 = U+0068 : LATIN SMALL LETTER H
69 = U+0069 : LATIN SMALL LETTER I
6A = U+006A : LATIN SMALL LETTER J
6B = U+006B : LATIN SMALL LETTER K
6C = U+006C : LATIN SMALL LETTER L
6D = U+006D : LATIN SMALL LETTER M
6E = U+006E : LATIN SMALL LETTER N
6F = U+006F : LATIN SMALL LETTER O
70 = U+0070 : LATIN SMALL LETTER P
71 = U+0071 : LATIN SMALL LETTER Q
72 = U+0072 : LATIN SMALL LETTER R
73 = U+0073 : LATIN SMALL LETTER S
74 = U+0074 : LATIN SMALL LETTER T
75 = U+0075 : LATIN SMALL LETTER U
76 = U+0076 : LATIN SMALL LETTER V
77 = U+0077 : LATIN SMALL LETTER W
78 = U+0078 : LATIN SMALL LETTER X
79 = U+0079 : LATIN SMALL LETTER Y
7A = U+007A : LATIN SMALL LETTER Z
7B = U+007B : LEFT CURLY BRACKET
7C = U+007C : VERTICAL LINE
7D = U+007D : RIGHT CURLY BRACKET
7E = U+007E : TILDE
7F = U+007F : DELETE
```

```
80 = U+20AC : EURO SIGN
81 = U+067E : ARABIC LETTER PEH
82 = U+201A : SINGLE LOW-9 QUOTATION MARK
83 = U+0192 : LATIN SMALL LETTER F WITH HOOK
84 = U+201E : DOUBLE LOW-9 QUOTATION MARK
85 = U+2026 : HORIZONTAL ELLIPSIS
86 = U+2020 : DAGGER
87 = U+2021 : DOUBLE DAGGER
88 = U+02C6 : MODIFIER LETTER CIRCUMFLEX ACCENT
89 = U+2030 : PER MILLE SIGN
8A = U+0679 : ARABIC LETTER TTEH
8B = U+2039 : SINGLE LEFT-POINTING ANGLE QUOTATION MARK
8C = U+0152 : LATIN CAPITAL LIGATURE OE
8D = U+0686 : ARABIC LETTER TCHEH
8E = U+0698 : ARABIC LETTER JEH
8F = U+0688 : ARABIC LETTER DDAL
90 = U+06AF : ARABIC LETTER GAF
91 = U+2018 : LEFT SINGLE QUOTATION MARK
92 = U+2019 : RIGHT SINGLE QUOTATION MARK
93 = U+201C : LEFT DOUBLE QUOTATION MARK
94 = U+201D : RIGHT DOUBLE QUOTATION MARK
95 = U+2022 : BULLET
96 = U+2013 : EN DASH
97 = U+2014 : EM DASH
98 = U+06A9 : ARABIC LETTER KEHEH
99 = U+2122 : TRADE MARK SIGN
9A = U+0691 : ARABIC LETTER RREH
9B = U+203A : SINGLE RIGHT-POINTING ANGLE QUOTATION MARK
9C = U+0153 : LATIN SMALL LIGATURE OE
9D = U+200C : ZERO WIDTH NON-JOINER
9E = U+200D : ZERO WIDTH JOINER
9F = U+06BA : ARABIC LETTER NOON GHUNNA
A0 = U+00A0 : NO-BREAK SPACE
A1 = U+060C : ARABIC COMMA
A2 = U+00A2 : CENT SIGN
A3 = U+00A3 : POUND SIGN
A4 = U+00A4 : CURRENCY SIGN
A5 = U+00A5 : YEN SIGN
A6 = U+00A6 : BROKEN BAR
A7 = U+00A7 : SECTION SIGN
A8 = U+00A8 : DIAERESIS
A9 = U+00A9 : COPYRIGHT SIGN
AA = U+06BE : ARABIC LETTER HEH DOACHASHMEE
AB = U+00AB : LEFT-POINTING DOUBLE ANGLE QUOTATION MARK
AC = U+00AC : NOT SIGN
AD = U+00AD : SOFT HYPHEN
AE = U+00AE : REGISTERED SIGN
AF = U+00AF : MACRON
B0 = U+00B0 : DEGREE SIGN
B1 = U+00B1 : PLUS-MINUS SIGN
B2 = U+00B2 : SUPERSCRIPT TWO
B3 = U+00B3 : SUPERSCRIPT THREE
B4 = U+00B4 : ACUTE ACCENT
B5 = U+00B5 : MICRO SIGN
B6 = U+00B6 : PILCROW SIGN
B7 = U+00B7 : MIDDLE DOT
B8 = U+00B8 : CEDILLA
B9 = U+00B9 : SUPERSCRIPT ONE
BA = U+061B : ARABIC SEMICOLON
BB = U+00BB : RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK
BC = U+00BC : VULGAR FRACTION ONE QUARTER
BD = U+00BD : VULGAR FRACTION ONE HALF
BE = U+00BE : VULGAR FRACTION THREE QUARTERS
BF = U+061F : ARABIC QUESTION MARK
```

C0 = U+06C1 : ARABIC LETTER HEH GOAL
C1 = U+0621 : ARABIC LETTER *HAMZA*
C2 = U+0622 : ARABIC LETTER ALEF WITH MADDA ABOVE
C3 = U+0623 : ARABIC LETTER ALEF WITH *HAMZA* ABOVE
C4 = U+0624 : ARABIC LETTER WAW WITH *HAMZA* ABOVE
C5 = U+0625 : ARABIC LETTER ALEF WITH *HAMZA* BELOW
C6 = U+0626 : ARABIC LETTER YEH WITH *HAMZA* ABOVE
C7 = U+0627 : ARABIC LETTER ALEF
C8 = U+0628 : ARABIC LETTER BEH
C9 = U+0629 : ARABIC LETTER TEH MARBUTA
CA = U+062A : ARABIC LETTER TEH
CB = U+062B : ARABIC LETTER THEH
CC = U+062C : ARABIC LETTER JEEM
CD = U+062D : ARABIC LETTER HAH
CE = U+062E : ARABIC LETTER KHAH
CF = U+062F : ARABIC LETTER DAL
D0 = U+0630 : ARABIC LETTER THAL
D1 = U+0631 : ARABIC LETTER REH
D2 = U+0632 : ARABIC LETTER ZAIN
D3 = U+0633 : ARABIC LETTER SEEN
D4 = U+0634 : ARABIC LETTER SHEEN
D5 = U+0635 : ARABIC LETTER SAD
D6 = U+0636 : ARABIC LETTER DAD
D7 = U+00D7 : MULTIPLICATION SIGN
D8 = U+0637 : ARABIC LETTER TAH
D9 = U+0638 : ARABIC LETTER ZAH
DA = U+0639 : ARABIC LETTER AIN
DB = U+063A : ARABIC LETTER GHAIN
DC = U+0640 : ARABIC TATWEEL
DD = U+0641 : ARABIC LETTER FEH
DE = U+0642 : ARABIC LETTER QAF
DF = U+0643 : ARABIC LETTER KAF
E0 = U+00E0 : LATIN SMALL LETTER A WITH GRAVE
E1 = U+0644 : ARABIC LETTER LAM
E2 = U+00E2 : LATIN SMALL LETTER A WITH CIRCUMFLEX
E3 = U+0645 : ARABIC LETTER MEEM
E4 = U+0646 : ARABIC LETTER NOON
E5 = U+0647 : ARABIC LETTER HEH
E6 = U+0648 : ARABIC LETTER WAW
E7 = U+00E7 : LATIN SMALL LETTER C WITH CEDILLA
E8 = U+00E8 : LATIN SMALL LETTER E WITH GRAVE
E9 = U+00E9 : LATIN SMALL LETTER E WITH ACUTE
EA = U+00EA : LATIN SMALL LETTER E WITH CIRCUMFLEX
EB = U+00EB : LATIN SMALL LETTER E WITH DIAERESIS
EC = U+0649 : ARABIC LETTER ALEF MAKSURA
ED = U+064A : ARABIC LETTER YEH
EE = U+00EE : LATIN SMALL LETTER I WITH CIRCUMFLEX
EF = U+00EF : LATIN SMALL LETTER I WITH DIAERESIS
F0 = U+064B : ARABIC FATHATAN
F1 = U+064C : ARABIC DAMMATAN
F2 = U+064D : ARABIC KASRATAN
F3 = U+064E : ARABIC FATHA
F4 = U+00F4 : LATIN SMALL LETTER O WITH CIRCUMFLEX
F5 = U+064F : ARABIC DAMMA
F6 = U+0650 : ARABIC KASRA
F7 = U+00F7 : DIVISION SIGN
F8 = U+0651 : ARABIC *SHADDA*
F9 = U+00F9 : LATIN SMALL LETTER U WITH GRAVE
FA = U+0652 : ARABIC SUKUN
FB = U+00FB : LATIN SMALL LETTER U WITH CIRCUMFLEX
FC = U+00FC : LATIN SMALL LETTER U WITH DIAERESIS
FD = U+200E : LEFT-TO-RIGHT MARK
FE = U+200F : RIGHT-TO-LEFT MARK
FF = U+06D2 : ARABIC LETTER YEH BARREE